PRESENTATION 3.2.4

N91-17042

FAULT DETECTION AND FAULT MANAGEMENT

FLIGHT ELEMENTS
FAULT DETECTION AND FAULT MANAGEMENT

WHITE PAPER

H. Lum, A. Patterson-Hine, J. T. Edge, D. Lawler

November 17, 1989

Background

Implementation of high-performance on-board intelligent computational systems is required to meet the requirements envisioned for NASA's current missions and the missions projected for the year 2000 era. Intelligent computational systems must be capable of: integrating, interpreting, and understanding sensor input information; correlating that information to the "world model" stored within its data base and understanding the differences, if any; defining, verifying, and validating a command sequence to merge the "external world" with the "internal world model"; and controlling the vehicle and/or platform to meet the scientific and engineering mission objectives. Critical to the implementation of such a system is an evolutionary approach taken to establish the baseline infrastructure for a real-time fault detection and fault management/reconfiguration system; the computational requirements for both ground mission operations and in-flight monitoring and operations will be highly dependent on the use of parallel and distributed computing in a fault-tolerant environment not totally dictated by the traditional implementation of triple redundancy. Decreases in mission operations costs while, at the same time, preserving the safety and reliability of flight missions, require a new and evolutionary approach to fault detection and fault management especially with the development and implementation of expert systems for system monitoring and advise. Currently, Mission Operations are left on their own to reverse-engineer the system designs to determine the consequences of failures on systems and functions, which involves a labor-intensive operation for both the design analysis and the mission operations support. Even the use of expert systems to automate failure analysis will not solve the problem of converting the systems schematics to the representation required by expert systems, nor will it provide the assurance that the software has been properly validated for mission critical use. The analysis of complex systems utilizing advanced automation and robotics must include an analysis of the software required by these systems as well as the hardware architecture. Techniques for modeling hardware systems and components need to be extended to represent the behavior of the software and to characterize the hardware/software interfaces. Traditional hardware fault management strategies such as hierarchical failure containment must also be applied to software components and addressed from an overall system fault management concept.

Concept

Fault management for an intelligent computational system must be developed using a "top-down" integrated engineering approach. Previous fault tolerant systems have been developed from a "bottom-up" approach, i.e., emphasizing the architecture required for a fault tolerant system rather than integrating the architecture with the overall mission requirements including the spacecraft design, ground and in-flight mission operations, and the design knowledge obtained from conceptual design through simulation, tests, integration, and flight operations. The proposed approach includes integrating the overall environment involving sensors and their as-

sociated data; design knowledge capture; operations; fault detection, identification, and reconfiguration; testability; causal models including digraph matrix analysis; and, overall performance impacts on the hardware and software architecture. Finally, the overall concept will be evaluated in testbeds simulating an operational environment to demonstrate technology readiness/feasibility for user transfer, establish user confidence in the technology , and validate the hardware/software architecture including the cost models for project implementation.

Objectives

Implementation of the concept to achieve a real-time intelligent fault detection and management system will be accomplished via the implementation of several major objectives which constitute the elements of the basic system infrastructure. These objectives are as follows:

a. Development of fault tolerant/FDIR requirements and specifications from a systems level which will carry through from conceptual design through implementation and mission operations. This element includes design data capture and acquisition throughout the life cycle of the project/mission. Figure 1, FT/RM Analysis Environment represents a realistic conceptual approach which will comply with the requirements of this objective.

b. Implementation of monitoring, diagnosis, and reconfiguration at all system levels providing the capability for unambiguous isolation of failures and integration of all systems aspects with mission maintenance support and operations.

c. Optimize system operations to manage degraded system performance through "top-down" system integration of all interacting elements with highest priority given to system availability through reconfiguration of hardware, software, and communications data networks, protocols, and interfaces.

d. Lower development and operations costs through the implementation of an intelligent real-time fault detection and fault management system including the development of an unified information management system (UNIS). UNIS will provide the capability for users to access the database at all levels independent of the skill level of the user thereby allowing real-time planning and scheduling consistent with program changes and deviations. Figure 2, FT/RM Analysis Process in the Space Station Program, is an example of an UNIS-type implementation to meet this objective.

Current Research Activities and Milestones

The proposed effort for Fault Detection and Fault Management will leverage current on-going activities currently being sponsored by the Office of Space Station Freedom, the Defense Advanced Research Projects Agency (DARPA), and the Office of Naval Research (ONR). As a result, the required technology development program to meet the proposed objectives will "inherit" the underlying basic research and development and will represent a cost-effective program. In addition, Space Station Freedom is already addressing some of the technology elements in its Advanced Technology Development Program and these efforts can also be applied to the development of the basic infrastructure of the Intelligent Fault Detection and Fault Management System using data obtained from existing Space Station testbeds and the Space Shuttle Program.

Early milestones which can be achieved are as follows:

CY-90: Review technology and investigate/define leveraging opportunities; define concept and develop integrated program technology development and integration plan

CY-91: Complete detailed definition of the integrated program plan and implementation of the supporting R & D technology base

CY-92: Test and evaluation of the integrated design strategies through simulations and testbed demonstrations. A proposed testbed demonstration is described below:

System concepts for software reliability and fault management will be validated using the advanced automated Space Station Freedom's Thermal Control System (TCS) jointly developed by ARC and JSC. Rapid prototyping capabilities and simulations will be conducted on ARC's TCS Research Testbed and system verification and validation will be conducted on JSC's TCS Testbed simulating the operational environment. System analysis will include both hardware and software. Techniques for modeling hardware systems will be extended to represent the behavior of the software and to characterize the hardware and software interfaces. Traditional hardware fault management strategies such as hierarchical failure containment will be applied to software components. The end product will be the demonstration of a fully automated, real-time fault management and control system utilizing advanced automation technologies and a system causal model for developing the criteria and evaluation of potential systems for implementation in a flight and/or operational environment. This demonstration will be a joint effort between ARC and JSC and will extend and leverage the original TCS effort sponsored jointly by OAST and OSS under the Systems Autonomy Demonstration Project (SADP) effort.

CY-93: Proof-of-Concept demonstration in an operational environment and optimization of the systems requirements document

CY-94 and beyond: Optimization of the hardware and software architectures to correct identified system design deficiencies, if any, and improve run-time performance. Initiate validation procedures for technology to be transferred to the user.

Key Researchers and Organizations

Current key researchers for the proposed effort are as follows:

Ames Research Center * : Dr. Ann Patterson-Hine (Point of Contact)
Dr. Henry Lum

Johnson Space Center * : J. T. Edge (Point of Contact)
Dennis Lawler

Langley Research Center: Chuck Meissner (Point of Contact)

Marshall Space Flight Center: David Weeks (Point of Contact)

* Major participants at the present time.

Note: The research and development collaboration for this effort will also utilize the on-going collaborative efforts with industry and academia. Figure 3 shows the collaborative research team which currently exists at Ames and can be leveraged to support the program.

Facilities

Key facilities are identified below and represent existing facilities which may be augmented to support the proposed effort:

Ames Research Center: Advanced Architectures Testbed, ALS/UNIS Testbed, and Space Station TCS Research Testbed. Figure 4 is an example of an existing testbed.

Johnson Space Center: Various Space Station and Space Shuttle Testbeds

Marshall Space Flight Center: SSM/PMAD and ECLSS Testbeds

No new facilities are required for this effort.

Candidate Benefiting Programs

The programs which will benefit from this effort include Space Shuttle, Space Station Freedom, NASA/AF Advanced Launch Systems (ALS), and the Lunar/Mars Missions. It is expected that Space Shuttle will be an early benefactor and the technologies transferred to the Space Shuttle environment will serve as the basic infrastructure for Space Station Freedom which will then be augmented to provide the additional required capabilities.

Major system needs which will be satisfied by this effort will be a decrease in the long-term mission operations costs through the development of a robust, intelligent fault detection and management system, higher quality decisions rendered during periods of uncertainty, and preservation of the "corporate knowledge" for long-life missions/projects. "Short-term" savings are not expected due to "up-front" implementation costs although efficiencies in personnel utilization for ground mission operations can be anticipated.

Technology Issues/Holes

Major technology issues/holes are as follows:

a. Validation methodologies for integrated knowledge-based systems (KBS) - Verification and validation (V&V) techniques are not tried (proven beyond doubt) and integrated for knowledge-based systems, i.e., systems that integrate both algorithmic and heuristic information. Validation processes are required before knowledge-based systems (also known as expert systems, intelligent systems, autonomous systems, and/or smart systems) will be incorporated into flight elements and in-line ground mission operations. Technical issues include: integration of validation processes; risk level permitted; applicable functional uses, i.e., critical and/or non-critical functions; languages; and validated software development tools.

b. Advanced integrated space-qualified multiprocessing architectures for intelligent fault detection, management, and control systems - Projected space-qualified architectures and processors do not address the hardware and software issues associated with highly automated fault

detection and management systems. This problem is increased when parallel processors, distributed processors, and knowledge-based systems are integrated into a heterogeneous computing environment. Issues include adaptive operating systems, languages; dynamic memory management and reallocation; network management; dynamic database management and consistency (truth maintenance); and validated on-chip testability functions.

c. Realistic causal model as the basis for automated fault detection, management, and control systems and general systems engineering analyses - A realistic causal model does not exist for the implementation of an automated (knowledge-based) fault detection, management, and control system and systems analysis. As a result, project managers cannot evaluate the effectiveness of automated systems. The automated Thermal Control System, jointly developed by Ames Research Center and Johnson Space Center, represents a start in the development of a realistic causal model. The effort has to be extended to reflect the entire system (only 25% of the system was automated for the Space Station Freedom engineering demonstration). Such a "core" model must support, in a principle manner, a broad range of systems engineering analysis such as: cost analysis, risk analysis, OPS analysis, FMEA, testability analysis, integration analysis, and automation analysis. (This concept is shown in Figure 1.)

d. Development and Maintenance of a reliability database - Reliability data is historically not available for NASA programs in a timely manner and is constrained by procurement procedures. Hence, NASA must develop the required databases using small samples which can be scalable.

e. Development of a theoretical foundation for systems engineering and integration - Only ad hoc techniques and techniques applicable to isolated systems and functions are currently available. A accepted general theory is not available to support the broad integrated analysis for the launch system as an entire system throughout the system lifecycle. Specific quantitative metrics are required for system engineers to accurately judge the consistency and completeness with which a current design meets systems requirements and constraints.

Recommended Cultural Changes

The following NASA cultural changes are recommended to facilitate this technology development:

a. Acceptance of fault detection, fault management, and control as an INTEGRATED SYSTEM ENGINEERING DISCIPLINE and not as a R&QA requirement, i.e., use a top-down integrated engineering approach.

b. Acceptance of fault management and control as a complementary approach to the classical (traditional) fault tolerant approach (triple redundancy). Maximize system availability with minimum system degradation.

c. Relaxation of validation requirements for knowledge-based systems, i.e., determination of an acceptable level of risk for systems incorporating heuristic (non-deterministic) information.

d. Incorporate systems engineering and integration as a driving force/organization in large complex system developments. Currently this discipline shares equal levels of design influence with areas such as OPS. This is inappropriate for driving the required functionality into the design while meeting other design constraints such as cost and fault tolerance.

FT/RM ANALYSIS ENVIRONMENT

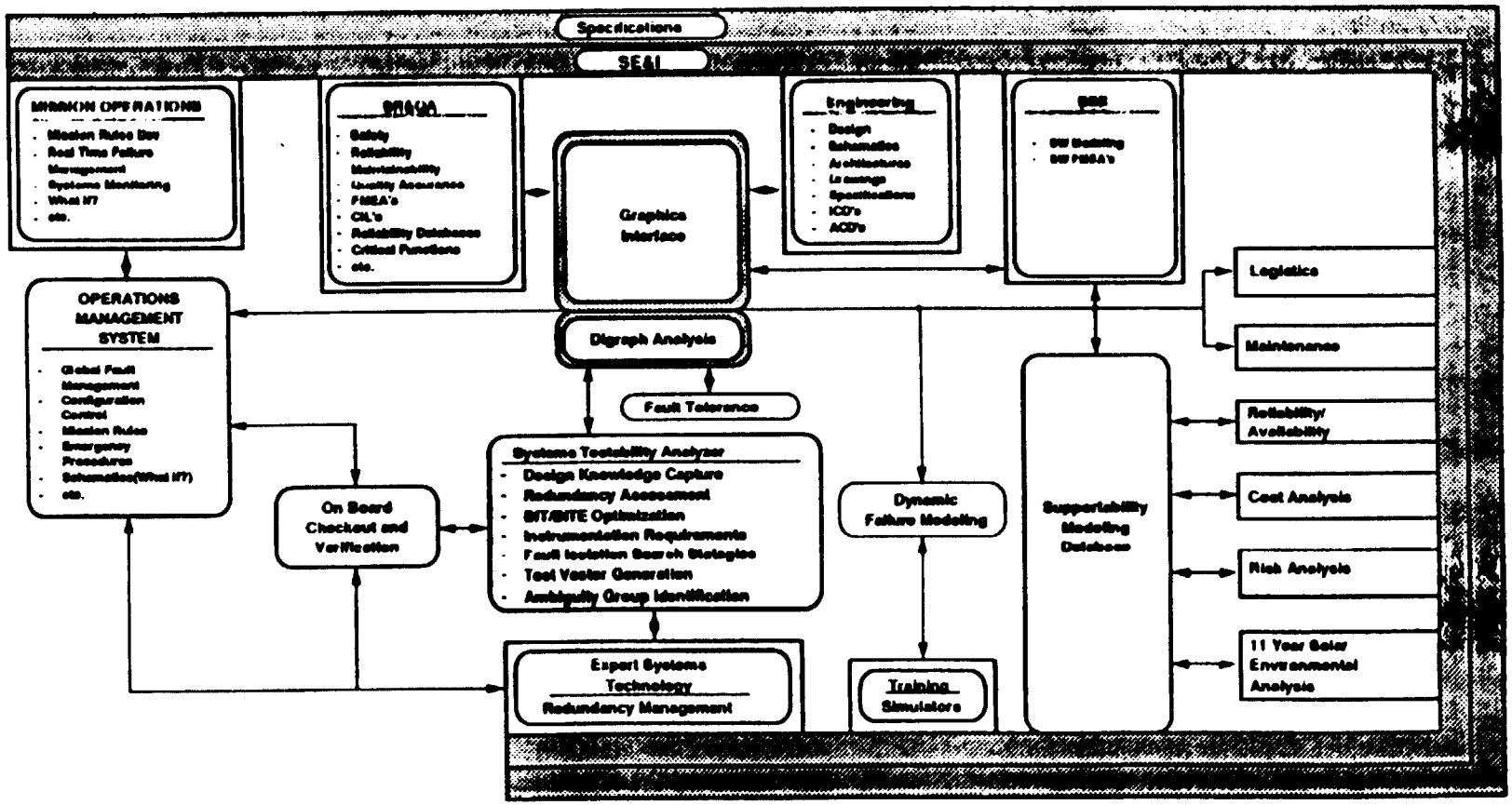

Figure 1

FT/RM ANALYSIS PROCESS
IN THE SPACE STATION PROGRAM

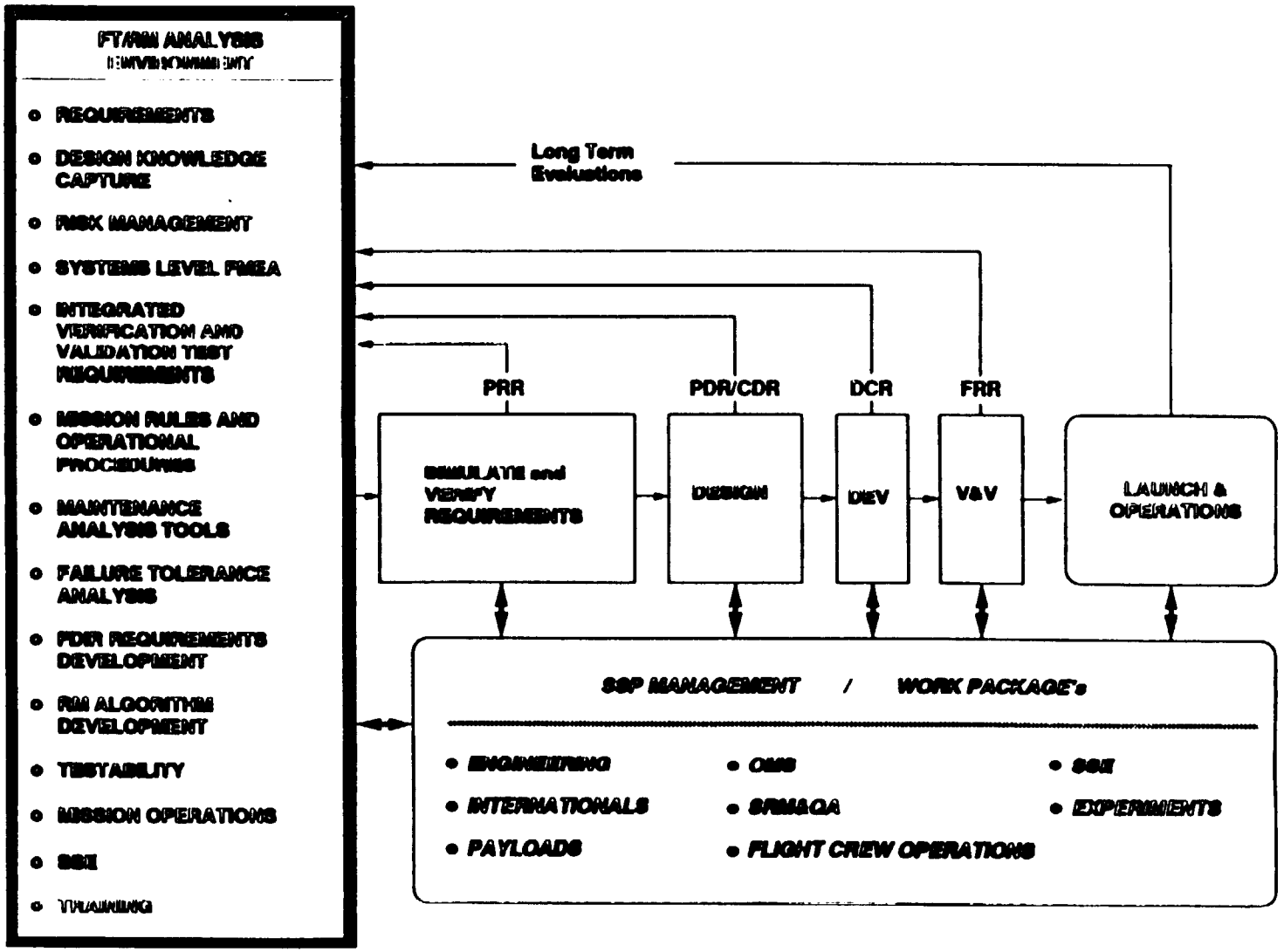

Figure 2

AMES COLLABORATIVE AI AND COMPUTER ARCHITECTURES RESEARCH TEAM

INFORMATION SCIENCES DIVISION (RI)

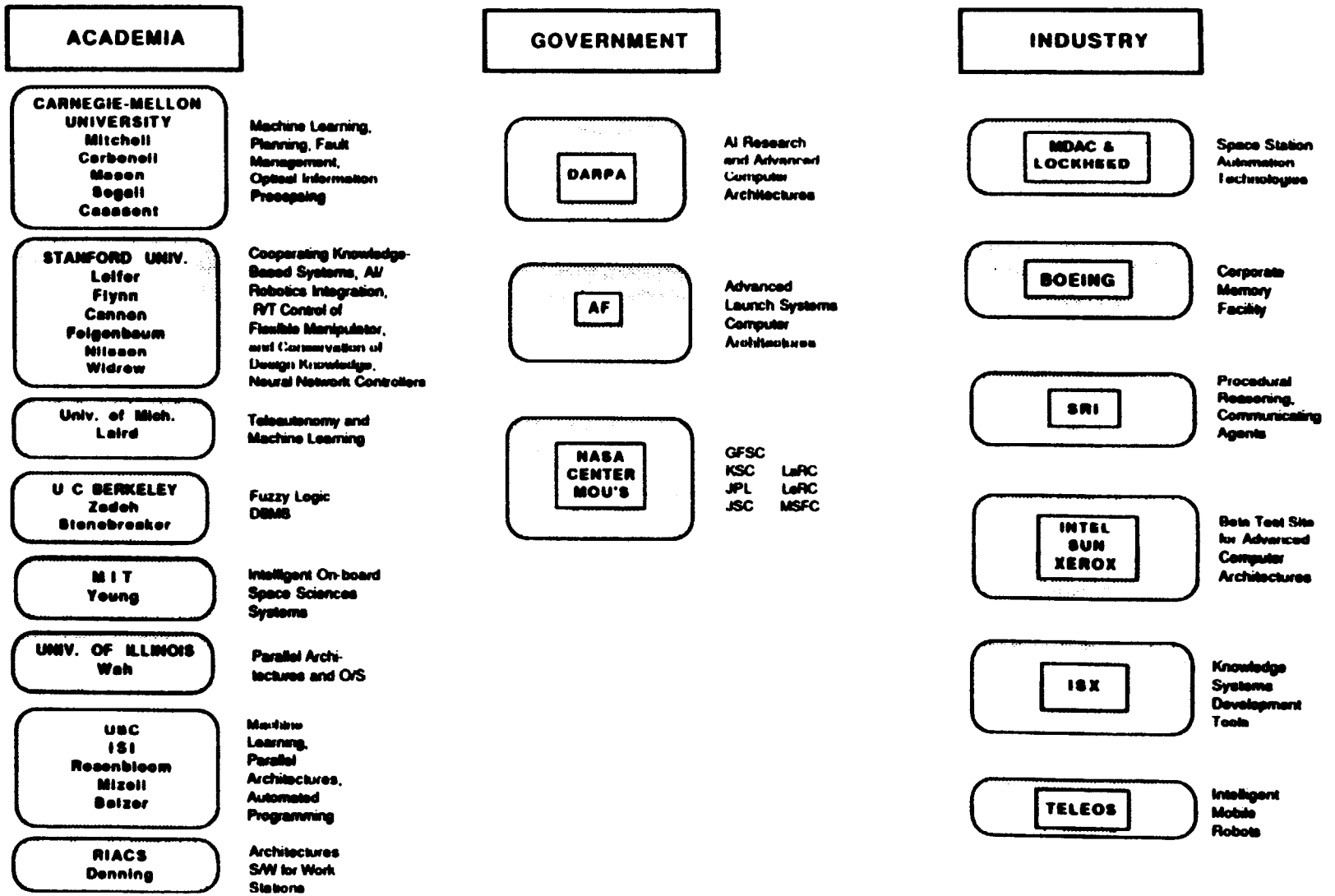

Figure 3

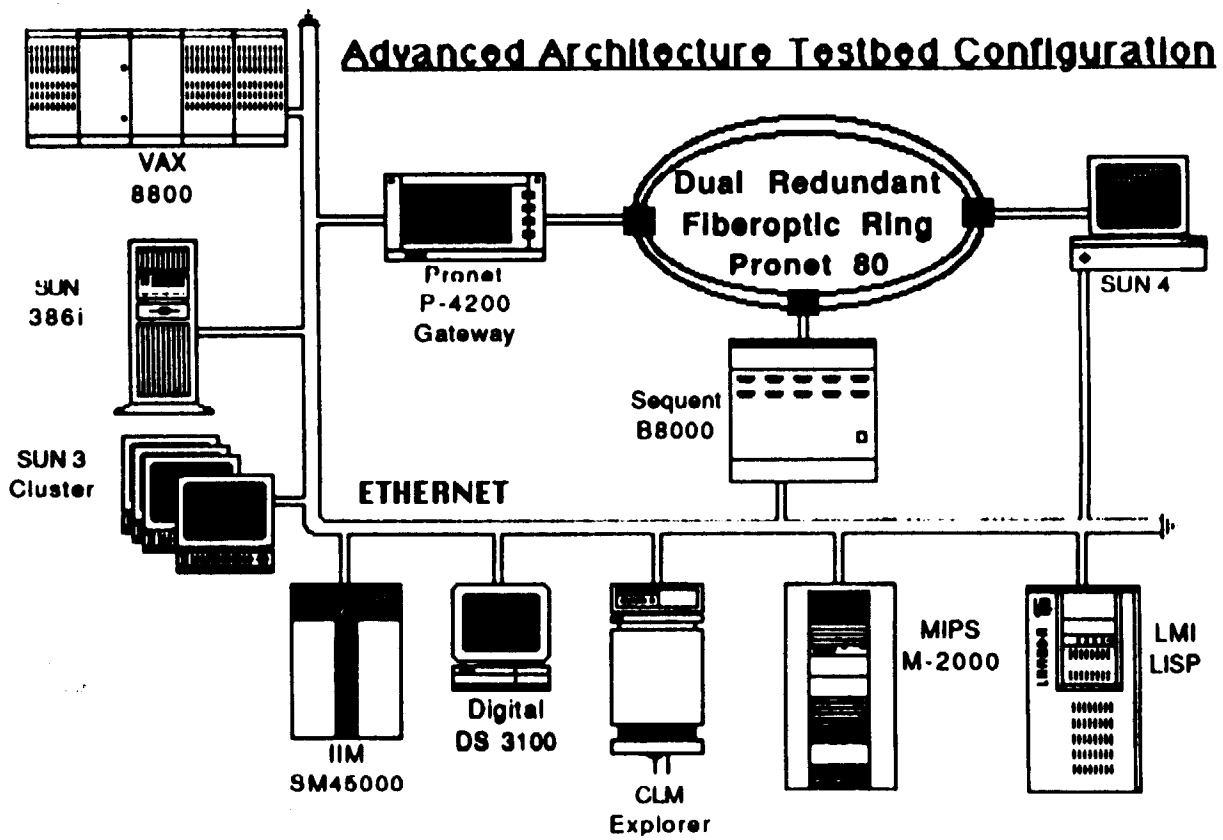
Advanced Architecture Testbed Configuration
VAX
8800
Pronet
P-4200
Gateway
Dual Redundant
Fiberoptic Ring
Pronet 80
SUN 4
SUN
386i
Sequent
B8000
SUN 3
Cluster
ETHERNET
IIM
SM45000
Digital
DS 3100
CLM
Explorer
MIPS
M-2000
LMI
LISP

Figure 4